

Review of the Paper: From Images to Shape Models for Object Detection

Vishal

Abstract

Object detection is used when we want to answer this particular query "Whether a given object occurs in an image and if it does, where is it?" This can be helpful when you know what you are looking for in the surroundings. In this review I will be discussing the approach followed in the paper "From Images to Shape Models for Object Detection" [1]

Outline of object detection

Any object detection algorithm may be divided into these three basic steps.

1. **Representation:** How are we trying to model the object and what are the features that we are planning to use to do so.
2. **Learning:** The machine learning algorithms that we apply to learn about the common class property of a particular object.
 - (a) **Supervised Algorithms:** In these type of algorithms we have to specify the location of the object in the training images which can be either by a bounding box or by the exact boundaries of the object.
 - (b) **Unsupervised Algorithms:** These kind of algorithms do not require bounding boxes in the training images and do not make any assumption about the area occupied by the object in the training images, but these may require some negative training samples as well.
3. **Recognition:** Identify the object in an image using models.

Approach of the paper

Training Set

This is a supervised algorithm and it requires test images with bounding boxes containing the objects.

Features

They use the Berkeley edge detector and approximately straight segments are used to fit the edges. The features used are *Pair of adjacent segments* (PAS) which is basically a pair of connected segments. A PAS feature $P = (x, y, s, e, d)$ has a location (x, y) (mean over the two segment centres), a scale s (distance between the segment centres), a strength e and a descriptor $d = (\theta_1, \theta_2, l_1, l_2, r)$ (the orientations, lengths and relative location vector) of the two segments forming the PAS. Two PAS may overlap.

The authors have said that the PAS features perform better because they can get rid of clutter easily as the chances of the same pattern occurring in the training images as clutter are very slim. [1, They have intermediate complexity, they are complex enough to be informative but simple enough to be detectable across different images and object instances]



The sample PAS features

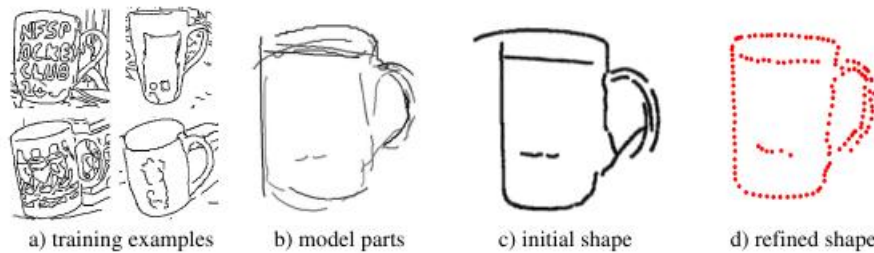
Forming the shape model

PAS codebook

This is generated by clustering all the PAS inside the training images according to their descriptors.

Learning the shape model

1. Determine which PAS features from the codebook occur the most, these are the model parts.
2. Assemble an initial shape by selecting a PAS feature for all the model parts. This is done while keeping in mind that the PAS matched to a model part and the model part itself should resemble each other and the fact that if 2 PAS are overlapping in the training images then the chances of them being overlapping in the initial model are high. An optimisation function is created keeping these two things in mind.
3. Refine the initial shape by matching it back to the training images using TPS-RPM [2]
4. Learn the intra class deformations using PCA and assuming each example shape as a point in a $2p - D$ (with p number of points on each shape) space.



The stages of the training

Object Detection

1. **Initialisation by hough voting:** Hough voting typically return 10-40 maxima in a cluttered image, but still it is helpful as it reduces the problem complexity because now we have to focus just on the regions returned by the hough voting scheme. Hough voting alone has been used in some systems designed for pedestrian detection.
2. **Shape matching by TPS-RPM:** The TPS-RPM (Thin Plate Spline Robust Point Matching) is basically a point matching algorithm here it is used to match the points between the regions returned by the hough voting scheme and the refined model of the class that we have with us. The matching is done keeping in mind that a point in the model should be matched as close as possible to its location in the test image and there should be no overfitting. The intra class deformations learnt during the training phase are used for constrained shape matching in this stage.

Datasets

1. **ETHZ shape classes:** A total of 255 images from the internet featuring 5 classes
2. **INRIA Horses:** 170 images with one or more horses and 170 images without any horses.

Results

One good thing about this algorithm is that it gives the boundary of the object and not just the bounding box as is done by most random fields based algorithms. The algorithm performs well on all classes except giraffes, the writers have not tried to explain this but maybe this is because of the huge variation possible in the orientations of the legs of a giraffe in the training images which leads to the formulation of an incorrect model.

When trying to localise object boundaries the coverage ranges from 78-92 % for all classes except giraffes with false positives at 0.4 FPPI (False Positive Per Image) which is the standard for comparing object

detection algorithms. Averaged over all classes it is 85.3 %. If the FPPI is set at 0.3 then the results decrease to 82.4 %.

Conclusion

All in all this is a very good algorithm and giving better results than all the past algorithms designed to do the same task. I will be making an attempt to make this algorithm unsupervised. In case of unsupervised algorithm the method to create the model of an object will be different than the paper presented, it will be along the lines of subgraph matching.

Bibliography

- [1] Vittorio Ferrari, Frédéric Jurie, and Cordelia Schmid. From images to shape models for object detection. *International Journal of Computer Vision*, 87(3):284–303, 2010.
- [2] H. Chui and A. Rangarajan. A new point matching algorithm for non-rigid registration. *Computer Vision and Image Understanding*, 89(2-3):114–141, 2003.